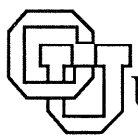
The PSMG Multiscale Method for an Anisotropic Problem

Yi-Hua Chang and Oliver A. McBryan

CU-CS-636-92 December 1992



University of Colorado at Boulder
DEPARTMENT OF COMPUTER SCIENCE

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
DEC 1992		**00-00-1992 to 00-00-1992**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
The PSMG Multiscale Method for an Anisotropic Problem	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Department of Computer Science,University of Colorado,Boulder,CO,80309**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**11**	

The PSMG Multiscale Method for an Anisotropic Problem*

Yi-Hua Chang , Oliver A. McBryan

Computer Science Department
University of Colorado
Boulder, CO 80309–0430

Abstract

The PSMG method provides an efficient solution method for elliptic PDE in the theoretically interesting limit where the number of processors approaches the number of discretization grid points. In this paper we show that the main features of PSMG may be extended from the Poisson equation on a square grid to anisotropic equations of similar type. Convergence rates of .12 or less per single-relaxation iteration are achieved for anisotropy coefficients ranging from 1 to over 10^5.

1 Introduction

This paper applies the PSMG multiscale method to solve an anisotropic equation. The PSMG multiscale algorithm [1,2,3] was introduced in 1986 by Frederickson and McBryan, as an efficient PDE solver for massively parallel architectures. Previous papers on PSMG have applied the method only to the Poisson equation on a uniform square grid. As is typical for point relaxation multigrid, PSMG will give very poor convergence if applied to an anisotropic equation such as:

$$-a\,\frac{\partial^2 U}{\partial x^2} - b\,\frac{\partial^2 U}{\partial y^2} = f(x,y)\ , \tag{1}$$

where a and b are positive constants with ratio significantly different from 1. For simplicity we will assume periodic boundary conditions.

Without loss of generality, we may transform equation (1) to the equation:

$$-\frac{\partial^2 U}{\partial x^2} - \epsilon\,\frac{\partial^2 U}{\partial y^2} = f(x,y)\ , \tag{2}$$

where $1 \le \epsilon < \infty$. We will refer to ϵ as the anisotropy parameter. We consider the equation (2) as our working model equation. We wish to solve (2) with periodic boundary conditions on a uniform grid. We use semi-coarsening methods, *i.e.* we descend to a coarse grid in one direction at a time in order to effectively cancel the anisotropy introduced by the ϵ term. We call these methods PSMGS and PSMGSS. The PSMGS method utilizes coarser grids only in the y direction, and can handle cases where $4 \le \epsilon < \infty$. The PSMGSS method alternately semi-coarsens in the x and y directions, and can handle cases where $1 \le \epsilon \le 4$.

*Research supported in part by AFOSR grant AFOSR-89-0422 and by NSF Grand Challenges Applications Grant ASC-9217394.

Throughout this paper we use the PSMG definitions and notations which are described in [1-3], as well as those for the regular Multigrid method, for example as in [4,5]. There are two parts to the description of our PSMG algorithms, corresponding to the grid space domain and the frequency domain. In the grid space domain we describe the implementation required on a computer in order to solve the equations. In the Fourier mode domain we describe the analysis required to get optimum coefficients for smoothing and interpolation. Section 2 presents the grid space analysis while section 3 presents the complex frequency domain analysis and choice of optimal coefficients. In section 4, we describe the convergence rates obtained for an hybrid solver based on PSMGS and PSMGSS.

2 Grid Space Domain

We will discretize equation (2) on a rectangle of sides X, Y by introducing an $n \times n$ periodic rectangular grid. The generalization to an $m \times n$ grid is trivial. For simplicity we assume that n is a power of 2: $n = 2^L$ and we therefore denote the grid by $G^{(L)}$. Using second order finite differences on $G^{(L)}$, the equation (2) becomes:

$$A^{(L)}U = f$$

where the operator $A^{(L)}$ has the usual 5-point form, and we include the superscript L to indicate its dependence on the grid domain.

The PSMG method requires an interpolation operator Q and a smoothing operator Z in addition to the difference operator A. Each of these will be chosen to involve only a small stencil of grid points, as with the 5-point A. Furthermore, since PSMG is a multiscale method, we will introduce a series of operators $A^{(l_x,l_y;L)}$, $Q^{(l_x,l_y;L)}$, and $Z^{(l_x,l_y;L)}$, all defined on $G^{(L)}$ but differing in their length scales - the integer distance offsets used in computing their stencils. The operators with superscript $(l_x, l_y; L)$ have scale lengths of $d(l_x)$ in the x direction and $d(l_y)$ in the y direction respectively, where $d(l) = 2^{L-l}$. These operators connect points that are a separation $d(l_x), d(l_y)$ apart on the periodic grid $G^{(L)}$.

All of these operators may be represented conveniently in the familiar "star" notation. In this notation we have:

$$A^{(l_x,l_y;L)} = \frac{1}{h_x^{(l_x)2}} \begin{bmatrix} & -\epsilon \frac{h_x^{(l_x)2}}{h_y^{(l_y)2}} & \\ -1 & 2\left(1 + \epsilon \frac{h_x^{(l_x)2}}{h_y^{(l_y)2}}\right) & -1 \\ & -\epsilon \frac{h_x^{(l_x)2}}{h_y^{(l_y)2}} & \end{bmatrix}^{(l_x,l_y;L)}$$

where $h_x^{(l)} = d(l)X/n$, $h_y^{(l)} = d(l)Y/n$) and where the superscript $(l_x, l_y; L)$ on the stencil tensor is a reminder that the operator is defined on the grid $G^{(L)}$ and uses scale distances $d(l_x)$ and $d(l_y)$ in the x and y directions respectively. The difference operator for our discretization of (2) is then simply the operator $A^{(L)} = A^{(L,L;L)}$.

In the PSMGS and PSMGSS method we will use only a 5-point star Q for the interpolation operator:

$$Q^{(l_x,l_y;L)} = \begin{bmatrix} & q_2 & \\ q_1 & q_0 & q_1 \\ & q_2 & \end{bmatrix}^{(l_x,l_y;L)}, \tag{3}$$

and a 1-point star (i.e. diagonal) Z for a smoothing operator:

$$Z^{(l_x,l_y;L)} = \left[\frac{z_0}{\frac{2}{h_x^{(l_x)2}} + \epsilon \frac{2}{h_y^{(l_y)2}}} \right]^{(l_x,l_y;L)} . \tag{4}$$

The parameters z_i and q_i will be chosen by optimization as described in section 3. We will define the PSMGS and PSMGSS methods in sections 2.1 and 2.2 respectively by defining the PSMG iteration operation $\mathbf{M}^{(L)}$ in terms of the operators A, Q and Z. Here $\mathbf{M}^{(L)}$ is the error reduction operator representing a complete PSMG cycle, defined in terms of the change in the error e of an approximate solution by: $e' = \mathbf{M}^{(L)}e$. $\mathbf{M}^{(L)}$ is defined by a recurrence equation that is given for PSMGS in section 2.1 and for PSMGSS in section 2.2. The PSMG convergence rate, μ, is defined by:

$$\mu = \sup_L \mu^{(L)} \quad , \quad \mu^{(L)} \equiv \| \mathbf{M}^{(L)} \| \tag{5}$$

and $\| \cdot \|$ denotes the operator norm on the space of grid-functions over $G^{(L)}$.

2.1 PSMGS:

In the PSMGS method, we semi-coarsen only in the y direction. As seen from the formula for $A^{(L,l;L)}$, the effect of the semicoarsening is to divide ϵ by 4 for each level of semi-coarsening. We will continue coarsening until we reach either level $L_{min} = 0$ or a level $L_{min} = L_{min}(\epsilon)$ which is the first level (i.e. integer) such that $L_{min} \geq L - log_4(\epsilon)$, whichever comes first. If $L_{min} = 0$ then we complete solution at that level by using one-dimensional solvers in the x direction. If $L_{min} > 0$ then the effective anisotropy coefficient is close to 1 - i.e. in the range $[1, 4]$ - and it is more appropriate to switch to a solver for near isotropic equations such as the PSMGSS method of the following section, or possibly the original PSMG method if the effective anisotropy is exactly 1.

We use a multiscale method as in PSMG, as opposed to a multigrid method. Thus all operators are defined with domain $G^{(L)}$. The basic steps involved at level l, $0 < l \leq L$, for the solution of $A^{(L)}U = f$, starting with an initial guess u, are described by:

Algorithm PSMGS(L, l, u, f):

0. If $l \equiv L_{min}$, exactly solve $A^{(L,l;L)}u = f$. Otherwise
1. Compute residual: $r = f - A^{(L,l;L)}u$.
2. Project residual to coarse grid: $r = r$ (trivial injection).
3. Solve coarse grid residual equation using PSMG: $e = PSMGS(L, l-1, 0, r)$.
4. Interpolate to fine grid: $e' = Q^{(L,l;L)}e$.
5. Apply a relaxation: $e'' = (I - Z^{(L,l;L)}A^{(L,l;L)})e' + Z^{(L,l;L)}r$.
6. Compute and return the new solution: $u'' = u + e''$.

To solve the original equation we would call PSMGS(L,L,u,f). The recurrence formula for the error reduction operator $\mathbf{M}^{(L)}$ in equation (5) is similar to PSMG (see [1]) and can be expressed in the following recurrence equation:

$$\mathbf{M}^{(L,l;L)} = \left(I - Z^{(L,l;L)}A^{(L,l;L)}\right)\left[I - A^{(L,l;L)}Q^{(L,l;L)}A^{(L,l-1;L)^{-1}}\left(I - \mathbf{M}^{(L,l-1;L)}\right)\right],$$

The required level L error reduction operator is then given by: $M^{(L)} = M^{(L,L;L)}$.

The exact solution required on level L_{min} consists of performing a 1-d solution in the x direction at each grid point. Because the original PDE and its discretization are singular due to translation invariance (assuming periodic boundary conditions), the exact solution required will encounter an indeterminacy, usually exhibited as a divide by zero. A simple resolution is to use a least squares solution. Indeed any approximate solver may be used at this point, with convergence rates depending on the solver used.

2.2 PSMGSS:

In the PSMGSS method we descend to a coarsest grid level in compound steps in which we coarsen by one level in the y direction, and then by one level in the x direction. As seen from the formula for $A^{(l_x,l_y;L)}$, the effect of the first semicoarsening is to divide ϵ by 4 while the second semi-coarsening then multiplies it by 4. So the PSMGSS method effectively maintains the value of ϵ as it descends the levels in steps of two. It provides a method similar in spirit to the original PSMG algorithms of [1-3], but allowing slightly non-isotropic problems to be handled as easily as the exactly isotropic case. Since the end-result of a PSMGS cycle is a near isotropic equation on the coarsest level L_{min}, one could use PSMGSS as an appropriate solution method for level L_{min}.

We introduce separate interpolation and smoothing operators Q_x, Z_x and Q_y, Z_y for the two semi-coarsening steps, each defined as in (3) and (4) but with coefficients $q_{0x}, q_{1x}, q_{2x}, z_{0x}$ and $q_{0y}, q_{1y}, q_{2y}, z_{0y}$.

Let d denote the coarsening direction which alternates between x and y. The basic steps involved at level l, $0 < l \leq L$, for the solution of $A^{(L)}U = f$, starting with an initial guess u, are described by:

Algorithm PSMGSS(L, d, l_x, l_y, u, f)**:**

0. If $l_x \equiv L_{min}$, exactly solve $A^{(l_x,l_x;L)}u = f$. Otherwise
1. Compute residual: $r = f - A^{(l_x,l_y;L)}u$.
2. Project residual to coarse grid: $r = r$ (trivial injection).
3. Define (d', l_x', l_y') as $(y, l_x - 1, l_y)$ or $(x, l_x, l_y - 1)$ according as $d = x$ or $d = y$.
4. Recursively solve coarse grid residual equation: $e = PSMGSS(L, d', l_x', l_y', 0, r)$.
5. Interpolate to fine grid: $e' = Q_d^{(l_x,l_y;L)}e$.
6. Apply a relaxation: $e'' = (I - Z_d^{(l_x,l_y;L)}A^{(l_x,l_y;L)})e' + Z_d^{(l_x,l_y;L)}r$.
7. Compute and return the new solution: $u'' = u + e''$.

To solve the original equation we would call PSMGSS(L,y,L,L,u,f). The recurrence formula for PSMGSS is then more complicated than for PSMG:

$$\mathrm{M}^{(l,l;L)} = \left(I - Z_y^{(l,l;L)}A^{(l,l;L)}\right) * \Big[I - Z_x^{(l,l-1;L)}Q_y^{(l,l;L)}A^{(l,l;L)} - \left(I - Z_x^{(l,l-1;L)}A^{(l,l-1;L)}\right) A^{(l,l-1;L)}Q_x^{(l,l-1;L)}Q_y^{(l,l;L)}A^{(l-1,l-1;L)^{-1}}\left(I - \mathrm{M}^{(l-1,l-1;L)}\right)\Big]$$

The error reduction operator is obtained from $M^{(L)} = M^{(L,L;L)}$.

The same comments apply to PSMGSS as were made for PSMGS about the need to take care of indeterminacy on level L_{min} for translation invariant problems. The most convenient

solution is to use a least squares definition of the singular solution on the bottom level.

3 Frequency Domain and Optimization (Fourier Mode Analysis)

In order to get a fast convergence rate, we will optimize the choice of PSMGS and PSMGSS coefficients in the interpolation and relaxation operators. This is most easily done in the frequency domain using Fourier mode analysis. This procedure is similar to that for PSMG although the operators are not the same.

For $i = 1, 2$ we introduce for convenience the trigonometric quantities:

$$x_i^{(l)} \;=\; \cos\left(2\pi k_i d(l)/n\right).$$

where we have omitted an explicit reference to the implicit dependence on L contained in $d(l)$, and where the allowed frequencies are given by $0 \le k_i < n$.

Using the discrete Fourier Transform on $G^{(L)}$ we obtain the frequency representation of the various operators as functions of the frequency parameters (k_1, k_2):

$$A^{(l_x,l_y;L)} = h_x^{(l_x)-2}\left[2(1-x_1^{(l_x)}) + 2\epsilon\frac{h_x^{(l_x)2}}{h_y^{(l_y)2}}(1-x_2^{(l_y)})\right],$$

$$Z^{(l_x,l_y;L)} = z_0/\left[\frac{2}{h_x^{(l_x)2}} + \epsilon\frac{2}{h_y^{(l_y)2}}\right],$$

$$Q^{(l_x,l_y;L)} = q_0 + 2(q_1 x_1^{(l_x)} + q_2 x_2^{(l_y)}).$$

Thus all of these operators are reduced to multiplication by simple polynomials in the $x_i^{(l)}$. The recurrence operator equations for $\mathbf{M}^{(l_x,l_y;L)}$ presented in sections 2.1 and 2.2 are now simply recurrence function equations of the same form.

3.1 PSMGS:

From section 2.1 we see that the recurrence relation for $\mathbf{M}^{(L,l;L)}$ involves the possibly singular quantity $A^{(L,l;L)}Q^{(L,l;L)}A^{(L,l-1;L)^{-1}}$. There are singular points of $A^{(L,l-1;L)^{-1}}$ at $(x_1, x_2) = (1, \pm 1)$. The singularity at (1,-1) is actually cancelled by the corresponding zero of $A^{(L,l;L)}$ at that point. We will choose Q so as to cancel the remaining singularity. This requires the following constraint on the coefficients of Q:

$$q_0 \;+\; 2q_1 \;-\; 2q_2 \;=\; 0.$$

The conventional PSMG constraint on $Q^{(L,l;L)}$ needed to ensure convergence near the origin in frequency space, independently of grid size (see [1]), requires:

$$q_0 \;+\; 2q_1 \;+\; 2q_2 \;= 1.$$

We may rewrite the above constraints as:

$$q_1 \;=\; 0.25\,(\,1\,-2q_0\,)\quad,\ \textit{and}\quad q_2 \;=\; 0.25\,.$$

As discussed in detail in [2], we can use a numerical optimization procedure on equation (5) to obtain the best choice of the remaining free coefficients q_0 and z_0, in order to get the fastest convergence rate. The PSMGS algorithm depends on the coefficient ϵ of the

anisotropy. We will continue coarsening until we reach a level $L_{min} = L_{min}(\epsilon)$ at which the effective anisotropy coefficient is close to 1 - i.e. such that $L - L_{min} = log_4(\epsilon)$. At that point it is more appropriate to switch to a solver for near isotropic equations - for example to the original PSMG solver. For the purpose of deriving optimal coefficients we will simply assume that an exact solution is performed on the level L_{min}.

For each ϵ, the optimization was in practice carried out for all levels L such that $L_{min} \leq L \leq 10$. Based on these tests for large L we have found that the optimal coefficients change very little when L is sufficiently large . As a representative example, the optimization program for $L = 6$ produced coefficients for Q and Z as shown in Table 3.1.1.

ϵ	L_{min}	z_0	q_0	$\mu^{(L)}$
4096	1	.5005	.5029	.02872
3136	1	.5007	.5036	.03552
2704	1	.5008	.5040	.03937
2304	1	.5010	.5045	.04414
1764	1	.5013	.5053	.05161
1296	1	.5020	.5060	.06431
1024	1	.5033	.5059	.07934
784	2	.5026	.5045	.03525
576	2	.5035	.5056	.04349
400	2	.5052	.5071	.05435
256	2	.5113	.5079	.07306
256	3	.5084	.5050	.02927
196	3	.5113	.5060	.03482
144	3	.5164	.5075	.04296
100	3	.5262	.5094	.05263
64	3	.5404	.5120	.06440
64	4	.5217	.5073	.02805
36	4	.5349	.5112	.04161
16	4	.5634	.5181	.06093
4	5	.5600	.5430	.06840

Table 3.1.1 PSMGS coefficients for L = 6

3.2 PSMGSS:

In the PSMGSS method we follow the same procedure as for PSMGS; however, the equations are more complicated. From section 2.2 we see that the recurrence relation for $\mathrm{M}^{(l,l;L)}$ involves the possibly singular quantity $A^{(l,l-1;L)}Q_x^{(l,l-1;L)}Q_y^{(l,l;L)}A^{(l-1,l-1;L)^{-1}}$.

As for PSMGS, in order to cancel the singular points $(x_1, x_2) = (\pm 1, \pm 1)$, and to obey the PSMG conventional constraint on Q at low frequency, we require the constraints:

$$
\begin{aligned}
(q_{0y} - 2q_{1y} + 2q_{2y}) (q_{0y} - 2q_{1x} + 2q_{2x}) &= 0; \\
(q_{0y} + 2q_{1y} - 2q_{2y}) (q_{0y} + 2q_{1x} + 2q_{2x}) &= 0; \\
(q_{0y} - 2q_{1y} - 2q_{2y}) (q_{0y} - 2q_{1x} + 2q_{2x}) &= 0; \\
(q_{0y} + 2q_{1y} + 2q_{2y}) (q_{0y} + 2q_{1x} + 2q_{2x}) &= 1.
\end{aligned}
$$

Combining the above 4 equations requires the following constraints on $Q_x^{(l)}$ and $Q_y^{(l)}$:

$$\begin{aligned} 2q_{1x} &= q_{0x} + 2q_{2x}, \\ 2q_{2y} &= q_{0y} + 2q_{1y}, \\ 2q_{2y}q_{1x} &= 1/16 . \end{aligned}$$

We again optimize the convergence rate (5) as a function of the coefficients q_{0x}, q_{0y}, q_{2y} , z_{0x} and z_{0y}. The optimization program produced the coefficients shown in Tables 3.2.1 and 3.2.2. In Table 3.2.1 we set $L = 6$ and $L_{min} = 5$ so that we are performing a two-grid method, with two semi-coarsening steps between the two grids. In Table 3.2.2 we have $L = 6$ and $L_{min} = 1$ so that we are completing a full PSMGSS sweep; the convergence rates are naturally somewhat worse than for the two-grid method.

ϵ	z_{0y}	z_{0x}	q_{0y}	q_{0x}	q_{2y}	$\mu^{(L)}$
4.00	.585	.853	.536	.737	.2319	.0655
3.61	.597	.825	.537	.725	.2273	.0661
3.24	.612	.828	.541	.727	.2253	.0668
2.89	.715	.842	.548	.767	.2179	.0630
2.56	.738	.808	.550	.789	.2128	.0613
2.25	.764	.769	.550	.806	.2054	.0586
1.96	.797	.730	.548	.837	.1956	.0561
1.69	.836	.731	.553	.859	.1867	.0549
1.44	.803	.709	.556	.867	.1813	.0571
1.21	.860	.717	.569	.862	.1735	.0601
1.00	.934	.677	.574	.865	.163	.0674

Table 3.2.1: PSMGSS coefficients for $L = 6$, $L_{min} = 5$

ϵ	z_{0y}	z_{0x}	q_{0y}	q_{0x}	q_{2y}	$\mu^{(L)}$
4.00	.614	.875	.531	.740	.227	.0746
3.61	.684	.893	.536	.752	.222	.0749
3.24	.701	.886	.539	.754	.220	.0715
2.89	.720	.863	.541	.756	.216	.0721
2.56	.745	.824	.542	.766	.211	.0763
2.25	.775	.786	.542	.782	.204	.0782
1.96	.810	.750	.540	.813	.194	.0775
1.69	.851	.714	.528	.885	.177	.0747
1.44	.794	.681	.520	.926	.168	.0739
1.21	.839	.675	.522	.944	.158	.0824
1.00	.909	.666	.534	.935	.150	.0949

Table 3.2.2: PSMGSS coefficients for $L = 6$, $L_{min} = 1$

4 Results - An Anisotropic Solver

We have combined PSMGS and PSMGSS to build an Anisotropic Solver. We built the solver to use the coefficient data for PSMGS and PSMGSS given in the tables in section 3. On the coarsest level we use a 1-d exact solver. This does not obviously provide the absolute optimal coefficients for every ϵ. Given a fixed ϵ one can of course run the optimization

program to get the best coefficients for that ϵ, resulting in a convergence rate that is generally much lower than those presented below.

The solver depends on two parameters, the grid size n and the anisotropy ϵ, and assumes for simplicity that $n = 2^L$ where L is an integer. The solver defines two major cases that depend on the value of an integer, i, where: $4^i \leq \epsilon < 4^{i+1}$.

Case 1: $i \geq L$ *i.e.* $\epsilon \geq n^2$. We use PSMGS on all coarse grid levels and on the coarsest level we use a 1-d exact solver.

Case 2: $i < L$ *i.e.* $\epsilon < n^2$. We use PSMGS for the first i levels and then use PSMGSS for the remaining levels.

As test data, we have taken u to be random and f to be either random or zero. The case $f = 0$ is particularly convenient as it allows iteration to continue down to machine accuracy rather than round-off accuracy.

We have measured performance for values of ϵ ranging from 1 to $2^{18} = 262144$, and grid sizes n ranging from 2^6 to 2^{10}. The worst convergence rate achieved was lower than .12. A subset of the results are shown in Table 4.1 for the range of values $4^5 \leq \epsilon < 4^6$ and $L = 6$, in order to demonstrate the effectiveness of the solver for values of ϵ between powers of 4.

ϵ	$\mu^{(L)}$	ϵ	$\mu^{(L)}$	ϵ	$\mu^{(L)}$	ϵ	$\mu^{(L)}$
1024	.119	1600	.089	2304	.058	3136	.055
1089	.118	1681	.089	2401	.058	3249	.050
1156	.101	1764	.083	2500	.059	3364	.049
1225	.101	1849	.085	2601	.055	3481	.050
1296	.102	1936	.067	2704	.052	3600	.050
1369	.089	2025	.068	2809	.065	3721	.052
1444	.088	2116	.070	2916	.058	3844	.051
1521	.087	2209	.058	3025	.051	3969	.051

Table 4.1: Hybrid Solver convergence rates for $L = 6$.

5 References

1. P.O. Frederickson and O.A. McBryan, "Parallel Superconvergent Multigrid", in: *Multigrid Methods: Theory, Applications and Supercomputing*, ed. S. McCormick, Math Applications Series, vol. 110, pp. 195-210, Marcel-Dekker Inc., New York, 1988.

2. P.O. Frederickson and O.A. McBryan,"Normalized Convergence Rates for the PSMG Method", SIAM Journal on Scientific and Statistical Computing, Vol. 12, No. 1, pp. 221-229, January 1991.

3. P.O. Frederickson and O.A. McBryan,"Recent Developments for the PSMG Multiscale Method", in *Multigrid Methods III*, eds. W. Hackbusch and U. Trottenberg, ISNM No. 98, Birkhauser Verlag, Basel 1991.

4. W. Hackbusch and U. Trottenberg (eds.): *Multigrid Methods. Proceedings of the conference held at Köln-Porz, November 23-27, 1981.* Lecture Notes in Mathematics Vol. 960, Springer, Berlin, 1982.

5. A. Brandt, : *Multigrid techniques 1984 guide with applications to fluid dynamics.* GMD-Studie Nr. 85, 1984.